AP
IFW

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of:

Daniel E. KNAUSS et al.

Serial No.: 10/024,051

Filed: December 21, 2001

For: DOCUMENT/MESSAGE MANAGEMENT

Group Art Unit: 2178

Examiner: Kristina B. Honeycut

Attorney Docket: 088305-0143

APPELLANT'S BRIEF UNDER 37 C.F.R. 1.192

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

The following is the Appellant's Brief, submitted under the provisions of 37 C.F.R. 1.192. The fee of $500 (large entity) required by 37 C.F.R. 1.17(c) is included in the accompanying check.

1. Real Party in Interest

The real party in interest is the assignee of record, GE Information Services, Inc.

2. Related Appeals and Interferences

There are no related appeals or interferences that will directly affect, be directly affected by, or have a bearing on the present appeal, that are known to appellant or the appellant's patent representative.

3. Status of Claims

The present appeal is directed to claims 1-20, i.e., all of the presently pending claims in this application.

4. Status of Amendments

Claims 1-15 were originally pending in the application. In response to a first Office Action, Appellant amended claims 1, 4, 5, 7, 8, 11, 12, 14 and 15, and added claims 16-20. In response to a second, final Office Action, Appellant made a minor amendment to claim 5 to correct a typographical error in that claim. The Advisory Action stated that the reply to the final Office Action would be entered for purposes of appeal. All of the presently pending claims are rejected over cited art of record.

5. Summary of the Invention

The present invention relates to document management for an Electronic Data Interchange (EDI).

As explained on page 2 of the specification, EDI is conveniently used in electronic commerce for the exchange of documents between trading partners in a transaction, whereby each company is required to ensure that they have tailored software programs that map between two types of data, one being EDI data and the other being data in the company's internal system format.

Figure 3 of the drawings shows one embodiment of the invention, in which a virtual document interface 210 is provided between applications 220, 230, and various types of messages or documents 240, 250, 260. Each application has a single link to the virtual document interface 210, whereby previously-defined mappings to variables in a virtual document of the virtual document interface 210 are utilized in order to populate a target document or message (e.g., a bill of sale including a list of products being sent out) from a source document or message (e.g., a purchase order that requests the products on the list of products).

Figures 4 and 5 show more details of the present invention, in which the virtual document interface 210 is utilized to map data from a source data model 420 to a target data model 430. The source data model 420 includes metadata and processing rules, whereby each metadata represents a particular data value in the source document, such as a document number or document type.

By way of the mappings such as shown in Figures 4 and 5, data from a source document is transferred to its appropriate positions in a target document. Figure 8 shows a mapping of a metadata ZEIAR of the source document to a metadata element BIG_04_324 in a first target document and a metadata element BEG_03_324 in a second target document, by virtue of a mapping with a variable named "Document No." in the virtual document 825. In a similar manner, the metadata ZEIAR of the source document is mapped to a metadata BEG_07_587 of the second target document and to a metadata BSN_06_640 of a third target document, by virtue of a mapping with a variable named "Document Type" in the virtual document 825.

6. Concise Statement Listing Each Ground of Rejection For Review

The issues on appeal are whether the Examiner erred in rejecting claims 1-3, 5, 8, 12 and 15 under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,557,780 to Edwards et al.; claims 4, 7, 11 and 14 under 35 U.S.C. § 103(a) as being unpatentable over Edwards et al. in view of U.S. Patent No. 6,738,975 to Yee al. and further in view of U.S. Patent No. 6,871,331 to Bloom et al.; claims 6, 9, 10 and 13 under 35 U.S.C. § 103(a) as being unpatentable over Edwards et al. in view of U.S. Patent Publication No. 2003/0135584 to Roberts; and claims 16-20 under 35 U.S.C. § 103(a) as being unpatentable over Edwards et al. in view of U.S. Patent No. 6,598,046 to Goldberg et al.

7. Argument

A. References Cited Against the Claims:

i. Edwards et al. (U.S. Patent No. 5,557,780):

Edwards et al. is directed to an Electronic Data Interchange System for Managing non-standard data, whereby incoming EDI data is converted and stored to an internal format for maintenance and application, and the system can re-compile the EDI data into an originator's format and retransmit the data. See Abstract of Edwards et al.

ii. Roberts et al. (U.S. Patent Publication No. 2003/0135584):

Roberts is directed to a network and apparatus for creating network services, in which a template can be created to produce code that may be utilized to create a web service, and in which the template has a list of features to create a model that generates the web service. See Abstract of Roberts et al.

iii. Yee et al. (U.S. Patent No. 6,738,975):

Yee et al. is directed to an extensible distributed enterprise application integrated system, in which interoperability between two or more applications can be defined. See Abstract of Yee et al.

iv. Bloom et al. (U.S. Patent No. 6,871,331):

Bloom et al. is directed to a waveform and data entry apparatus, which automates the entry, modification, analysis and generation of test benches from electrical circuits, both of which are defined by hardware description language (HDL) files. See Abstract of Bloom et al.

v. Goldberg et al. (U.S. Patent No. 6,598,046):

Goldberg et al. is directed to a system for retrieving documents responsive to a given user's role and scenario, in which selection criteria are compared to metadata of each document to determine if there is a document match. See Abstract of Goldberg et al.

B. Claims 1-3, 5, 8, 12 and 15

Figure 15 of Edwards, as explained in column 11, lines 51-67 of Edwards, describes a methodology by which a class of transaction is automatically mapped into an EDI standard of another class of transaction. For example, a purchase order transaction is automatically mapped into a purchase order acknowledgement. To do this, mapping of segments is made from an EDI standard format into a new format, according to an EDI document cross-reference file 154 as seen in Figure 15 of Edwards.

The above-referenced portion of Edwards does not disclose or suggest the use of maps for mapping metadata of source documents to variables of a virtual document. Rather, it discloses the mapping of portions (or segments) of a source document in one format to portions (or segments) of a target document in another format. No discussion of metadata or variables of a virtual document is disclosed or suggested by Edwards. In fact, the term "metadata" is not mentioned in Edwards, since it does not utilize metadata. Rather, Edwards determines 'segments' of a source document based on segment terminators and element separator characters in a source document, and operates on those segments, as described in column 8, lines 39-44 of Edwards.

As mentioned above, Edwards provides a system that creates another document, such as a Purchase Order Acknowledgement, automatically from a document that has been received, such as a Purchase Order sent from a client. Edwards does not perform any mapping from metadata of a source document to variables of a virtual document, and Edwards does not perform any mapping from variables of the virtual document to a target document. Rather, Edwards does a direct conversion of a source document to a target document, based on "segment mapping" of the source document to the target document.

The fact that Edwards teaches loading maps for mapping, as mentioned on page 3 of the final Office Action, does not meet the specific claim features recited in claim 1, whereby metadata of different types of source documents are mapped to variables of a virtual document, and whereby the variables of the virtual document are mapped to metadata of a particular type of target document.

Claim 1 specifically recites the steps of:

obtaining a first map from the plurality of maps based on the determined type of the source data model, and mapping the metadata of the source data model to corresponding variables of a virtual document in accordance with the first map;

determining a type of a target data model to which data from the source data model is to be transferred to; and

obtaining a second map from the plurality of maps based on the determined type of the target data model, and mapping the variables of the virtual document to metadata of a target data model having a second EDI format in accordance with the second map (emphasis added).

The Advisory Action points to column 6, lines 13-19 of Edwards for teaching mapping one document type to another document type, but this portion of Edwards does not teach or suggest the use of metadata in source and target documents and variables in a virtual document to accomplish such mappings. The Advisory Action further points to column 7, lines 15-21 of Edwards for teaching mapping of a document to a target document, whereby this portion of Edwards merely describes that a transaction is converted into an intended recipient trading partner's native standard format for return transmission via an event driven communications server 31. Again, this does not teach or suggest the specific features recited in independent claim 1.

Independent claims 5, 8, 12 and 15 recite similar features to those mentioned above with respect to independent claim 1.

Accordingly, claims 1-3, 5, 8, 12 and 15 are not anticipated by Edwards.

C. Claim 2

Claim 2 depends from claim 1, and so the comments provided above with respect to claim 1 apply equally as well to claim 2. In addition, claim 2 is patentable for these other reasons.

Claim 2 recites that a source message or document is translated to obtain its corresponding metadata. It is noted that Figure 2 of Edwards discloses an inbound transaction translator 32, which is also described in more detail in Figure 3 of Edwards. However, as described in column 4, lines 58-67 of Edwards, the inbound transaction translator 32 determines which EDI standard the incoming data relies on, and then translates the data from the incoming format to an in-house standard format called "internal system format", which can be used to display, print, store or modify the transaction. Such a format-to-format translator does not correspond to the translation of a source message or document to obtain its corresponding metadata, since no format translation to obtain metadata is being made in that instance, but rather an extraction process is being performed.

Accordingly, claim 2 is patentable for this additional reason, beyond the reasons provided above with respect to its base claim 1.

D. Claims 4, 7, 11 and 14:

Claim 4 recites that the variables of the virtual document are assigned semantic names representative of a type of data to be stored to the variables (e.g., "Document No." for document number), and in which the maps are created by a user prior to receiving the source data model, based on an intuitive correspondence made by the user from a particular metadata name of the source data model and a particular semantic name of a variable of the virtual document. Claims 7, 11 and 14 recite similar features.

In its rejection of claims 4, 7, 11 and 14 based on the combination of Edwards, Yee and Bloom, the final Office Action asserts that Figure 4 and column 8, lines 15-27 of Edwards discloses that variables of a virtual document are assigned semantic names representative of a type of data to be assigned to the variables. Appellant respectfully disagrees with this assertion made in the final Office Action. Column 8, lines 15-27 of Edwards merely describes that segments and their order in a document are determined, and stored in a segment name table file 62, whereby a type of document can then be determined. Thus, if a received document has a particular segment structure that is similar to a purchase order, it is determined to be a purchase order (e.g., has a particular product order segment structure). This has nothing at all to do with the assignment of semantic names representative of a type of data to variables of a virtual document.

Furthermore, the final Office Action asserts that Bloom teaches mapping based on data and names, in column 3, lines 42-45 and column 4, lines 59-61 of Bloom. Appellant respectfully disagrees, with respect to the specific features recited in claims 4, 7, 11 and 14. In particular, as recited in these claims, a user creates a map based on an intuitive correspondence made by the user from a particular metadata name of a source data model and a particular semantic name of a variable of the virtual document. At best, Bloom, which is directed to a totally different area of a combined data waveform and data entry apparatus for facilitating fast behavioral verification of digital hardware designs, performs automatic signal name remapping based on their name, mode and data type. No comparison (and certainly no intuitive comparison) is made by a user, or by a computer for that matter, in the system of Bloom, with respect to semantic names of virtual document variables and metadata names of a source data model.

Accordingly, claims 4, 7, 11 and 14 are patentable over the combined teachings of Edwards, Bloom, and Yee, for the reasons provided above, beyond those provided in the earlier sections of this Appeal Brief.

E. Claims 16-20

The final Office Action relies on Goldberg for teaching the features recited in claims 16-20, whereby Appellant respectfully disagrees with the final Office Action's understanding of that reference.

Claim 16 recites obtaining, from a database, source-independent data that is used to provide predetermined values for at least one of the metadata of the target data model (emphasis added). Claims 17-20 recite similar features.

In Goldberg, metadata tags and values of source documents are compared to predetermined metadata tags and values stored in a database, in order to extract documents of interest. Thus, if seven documents are stored in the system and only two of those documents correspond to personal taxpayer documents, the system of Goldberg can extract those two desired documents. This has nothing at all with providing predetermined values for at least one of the metadata of a target data model, as recited in claims 16-20, but rather it merely discloses the comparison of metadata values of source documents with metadata values stored in a database, whereby no inputting of the stored metadata values into a target document is taught or suggested by Goldberg.

Accordingly, since Edwards does not rectify the above-mentioned shortcomings of Goldberg, claims 16-20 are patentable over the combination of Edwards and Goldberg.

8. Summary

For the foregoing reasons, it is submitted that the Examiner's rejections are erroneous, and reversal of the applied rejections is respectfully requested.

Respectfully submitted,

Phillip J. Articola

Phillip J. Articola
Registration No. 38,819

Foley & Lardner LLP
3000 K Street, N.W., Suite 500
Washington, D.C. 20007-5109
Telephone: (202) 672-5300
Facsimile: (202) 672-5399

Date: September 27, 2005

10. Appendix:

Presently Pending Claims:

1. (Previously Presented) A computer implemented method of automatically generating Electronic Data Interchange (EDI) documents or messages using an EDI system, comprising:

storing a plurality of maps for respectively mapping metadata from different types of source documents to variables of a virtual document;

receiving a source data model having a first EDI format corresponding to EDI related data, the source data model including metadata;

determining a type of the source data model from the metadata;

obtaining a first map from the plurality of maps based on the determined type of the source data model, and mapping the metadata of the source data model to corresponding variables of a virtual document in accordance with the first map;

determining a type of a target data model to which data from the source data model is to be transferred to;

obtaining a second map from the plurality of maps based on the determined type of the target data model, and mapping the variables of the virtual document to metadata of a target data model having a second EDI format in accordance with the second map.

2. (Original) The method according to claim 1, wherein, when a source message or document is inputted to the EDI system, the source message or document is translated to obtain its corresponding metadata, and the values corresponding to the metadata are provided to the corresponding mapped variables of the virtual document at run time, and

wherein the corresponding values of the mapped variables of the virtual document are provided to the corresponding metadata of the target data model, so as to populate the target data model with data from the source data model.

3. (Original) The method according to claim 2, wherein the first EDI format is a data transaction formatting standard, and the second EDI format is a data transaction formatting standard.

4. (Previously Presented) The method according to claim 2, wherein the variables of the virtual document are assigned semantic names representative of a type of data to be stored to the variables, and

wherein the maps are created by a user prior to receiving the source data model, based on an intuitive correspondence made by the user from a particular metadata name of one of the metadata of the source data model and a particular semantic name of one of the variables of the virtual document.

5. (Previously Presented) A system for automatically generating data in a self-describing markup language format from EDI data, comprising:

a storing unit that stores a plurality of maps for respectively mapping metadata from different types of source documents to variables of a virtual document;

a receiving unit that receives a message or document from a first trading partner as EDI data;

a determining unit that determines a type of the message or document received from the metadata and that determines a type of a target data model;

a virtual document that obtains a first map from the plurality of maps stored in the storing unit based on the type of the message or document as determined by the determining unit, and that maps metadata from the message or document of the first trading partner to variables of the virtual document in accordance with the first map, and that obtains a second map from the plurality of maps stored in the storing unit based on the type of the target data model as determined by the determining unit, and that maps metadata from a message or document of a second trading partner to the variables of the virtual document in accordance with the second map; and

a transmitting unit that transmits values provided to the variables of the virtual document from the message or document from the first trading partner, to the corresponding metadata of the message or document of the second trading partner.

6. (Original) The system according to claim 5, wherein self-describing markup language format is XML.

7. (Previously Presented) The system according to claim 5, wherein the variables of the virtual document are assigned semantic names representative of a type of data to be stored to the variables, and

wherein the maps are created by a user prior to receiving the message or document from the first trading partner, based on an intuitive correspondence made by the user from a particular metadata name of one of the metadata of the message or document received from the first trading partner and a particular semantic name of one of the variables of the virtual document.

8. (Previously Presented) A computer readable data storage medium for an EDI system having program code recorded thereon that is executable by a computer to perform the following steps :

storing a plurality of maps for respectively mapping metadata from different types of source documents to variables of a virtual document;

receiving a source data model having a first EDI format corresponding to EDI related data, the source data model including metadata;

determining a type of the source data model from the metadata;

obtaining a first map from the plurality of maps based on the determined type of the source data model, and mapping the metadata of the source data model to corresponding variables of a virtual document in accordance with the first map;

determining a type of a target data model to which data from the source data model is to be transferred to;

obtaining a second map from the plurality of maps based on the determined type of the target data model, and mapping the variables of the virtual document to metadata of a target data model in accordance with the second map,

wherein, when a source message or document is received by the EDI system, the program code is programmed to:

translate the source message or document to obtain its corresponding metadata;

provide the values corresponding to the metadata to the corresponding mapped variables of the virtual document; and

provide the corresponding values of the mapped variables of the virtual document to the corresponding metadata of the target data model.

9. (Original) The computer readable data storage medium according to claim 8, wherein the EDI-related data is in a self-describing markup language format.

10. (Original) The computer readable data storage medium according to claim 9, wherein the self-describing markup language format is XML.

11. (Previously Presented) The computer readable data storage medium according to claim 8, wherein the variables of the virtual document are assigned semantic names representative of a type of data to be stored to the variables, and

wherein the maps are created by a user prior to receiving the source data model, based on an intuitive correspondence made by the user from a particular metadata name of one of the metadata of the source data model and a particular semantic name of one of the variables of the virtual document.

12. (Previously Presented) A system for automatically generating data in a self-describing markup language format from received EDI data, comprising:

storing means for storing a plurality of maps for respectively mapping metadata from different types of source documents to variables of a virtual document;

receiving means for receiving a message or document from a first trading partner as EDI data;

determining means for determining a type of the message or document received by the receiving means from the metadata, and for determining a type of a target data model;

a virtual document that obtains a first map from the plurality of maps stored in the storing means based on the type of the message or document received by the receiving means as determined by the determining means, and that maps metadata from the message or document of the first trading partner to variables of the virtual document in accordance with the first map, and that obtains a second map from the plurality of maps stored in the storing means based on the type of the target data model as determined by the determining means, and that maps metadata from a message or document of a second trading partner to the variables of the virtual document in accordance with the second map; and

transmitting means for transmitting values provided to the variables of the virtual document from the message or document from the first trading partner, to the corresponding metadata of the message or document of the second trading partner.

13. (Original) The system according to claim 12, wherein self-describing markup language format is XML.

14. (Previously Presented) The system according to claim 12, wherein the variables of the virtual document are assigned semantic names representative of a type of data to be stored to the variables, and

wherein the maps are created by a user prior to receiving the source data model, based on an intuitive correspondence made by the user from a particular metadata name of one of the metadata of the message or document received by the receiving means and a particular semantic name of one of the variables of the virtual document.

15. (Previously Presented) A method for automatically generating data in a prescribed format from a received EDI document or message having metadata elements, comprising:

assigning, by a first user, a first plurality of maps from metadata elements of different types of source documents to variables of a virtual document;

assigning, by a second user, a second plurality of maps from the variables of the virtual document to metadata elements of a target EDI document or message;

pulling values assigned to the metadata elements of the received EDI document or message to the variables of the virtual document, based on a source document-to-virtual document mapping that corresponds to one of the first plurality of maps that is automatically determined based on a type of the metadata elements of the received EDI document or message; and

pushing values assigned to the variables of the virtual document to metadata elements of the target EDI document or message, based on a target document-to-virtual document mapping that corresponds to one of the second plurality of maps that is automatically determined based on a type of the metadata elements of the target EDI document or message.

16. (Previously Presented) The method according to claim 1, further comprising:

obtaining, from a database, source-independent data that is used to provide predetermined values for at least one of the metadata of the target data model.

17. (Previously Presented) The system according to claim 5, further comprising:

a database that is communicatively connected to the virtual document, the database storing source-independent data that is used by the virtual document to provide predetermined values for at least one of the metadata of the target data model.

18. (Previously Presented) The computer readable data storage medium according to claim 8, wherein the program code is further programmed to:

obtain, from a database, source-independent data that is used to provide predetermined values for at least one of the metadata of the target data model.

19. (Previously Presented) The system according to claim 12, further comprising:

a database that is communicatively connected to the virtual document, the database storing source-independent data that is used by the virtual document to provide predetermined values for at least one of the metadata of the target data model.

20. (Previously Presented) The method according to claim 15, wherein at least one of the values pulled to the variables of the virtual document is obtained from source-independent information obtained from a database.